REFERENCES

[1] J. A. Mandelman, R. H. Dennard, G. B. Bronner, J. K. DeBrosse, R. Divakaruni, Y. Li, and C. J. Radens, "Challenges and future directions for the scaling of dynamic random-access memory (DRAM)," *IBM J. Res. Develop.*, vol. 46, nos. 2–3, pp. 187–212, Mar. 2002.

[2] P. Mazumder and K. Chakraborty, *Testing and Testable Design of High-Density Random-Access Memories*, 1st ed. Norwell, MA, USA: Kluwer, Sep. 1996.

[3] E. Gizdarski, "Built-in self-test for folded bit-line Mbit DRAMs," *Integr., VLSI J.*, vol. 21, nos. 1–2, pp. 95–112, Nov. 1996.

[4] H.-Y. Yang, C.-M. Chang, M. C. Chao, R.-F. Huang, and S.-C. Lin, "Testing methodology of embedded DRAMs," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 20, no. 9, pp. 1715–1727, Sep. 2012.

[5] M. C.-T. Chao, H.-Y. Yang, R.-F. Huang, S.-C. Lin, and C.-Y. Chin, "Fault models for embedded-DRAM macros," in *Proc. 46th ACM/IEEE Design Autom. Conf.*, Jul. 2009, pp. 714–719.

[6] Z. Yang and S. Mourad, "Crosstalk in deep submicron DRAMs," in *Proc. IEEE Int. Workshop Memory Technol., Design Test.*, Aug. 2000, pp. 125–129.

[7] W. Jeong, I. Kim, S. Lee, O. Kwon, E. Kal, K. Lee, H. Bae, K. Lee, and S. Kang, "A novel screen-ability estimation methodology for DRAM with a test algorithm simulator: FS5," in *Proc. Int. Tech. Conf. Circuits/Syst., Comput. Commun.*, 2010, pp. 927–929.

[8] A. J. van de Goor, *Testing Semiconductor Memories-Theory and Practice*. New York, NY, USA: Wiley, 1991.

[9] D.-C. Kang, S. M. Park, and S.-B. Cho, "An efficient built-in self test algorithm for neighborhood pattern and bit-line-sensitive faults in high density memories," *ETRI J.*, vol. 26, no. 6, pp. 520–534, 2004.

[10] M. Franklin, K. Saluja, and K. Kinoshita, "A built in self test algorithm for row/column pattern sensitive faults in RAMs," *IEEE J. Solid-State Circuits*, vol. 25, no. 2, pp. 514–524, Apr. 1990.

[11] S. Banerjee, D. R. Chowdhury, and B. B. Bhattacharya, "A programmable built-in self-test for embedded DRAMs," in *Proc. IEEE Int. Workshop Memory Technol. Design Test.*, Aug. 2005, pp. 58–63.

[12] J. Y. Kim, S. J. Hong, and J. Kim, "Parallely testable design for detection of neighborhood pattern sensitive faults in high density DRAMs," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2005, pp. 5854–5857.

[13] Y.-J. Huang and J.-F. Li, "Testing active neighborhood pattern-sensitive faults of ternary content addressable memories," in *Proc. IEEE Eur. Test Symp.*, May 2006, pp. 55–60.

[14] K.-L. Cheng, M.-F. Tsai, and C.-W. Wu, "Neighborhood pattern sensitive fault testing and diagnostics for random access memories," *IEEE Trans. Comput. Aided Design Integr. Circuits Syst.*, vol. 21, no. 11, pp. 1328–1336, Nov. 2002.

[15] S. Hamdioui and J. D. Reyes, "New data-background sequences and their industrial evaluation for word-oriented random-access memories," *IEEE Trans. Comput. Aided Design Integr. Circuits Syst.*, vol. 24, no. 6, pp. 892–904, Jun. 2005.

[16] Y. Sfikas and Y. Tsiatouhas, "Physical design oriented DRAM neighborhood pattern sensitive fault testing," in *Proc. IEEE Symp. Design Diag. Electron. Circuits Syst.*, Apr. 2009, pp. 108–113.

[17] B. Keeth and R. J. Baker, *DRAM Circuit Design: A Tutorial*, 1st ed. Piscataway, NJ, USA: IEEE Press, 2001.

[18] Y. Matsubara, M. Habu, S. Matsuda, K. Honda, E. Morifuji, T. Yoshida, K. Kokubun, K. Yasumoto, T. Sakurai, T. Suzuki, J. Yoshikawa, E. Takahashi, K. Hiyama, M. Kanda, R. Ishizuka, M. Moriuchi, H. Koga, Y. Fukuzaki, Y. Sogo, H. Takahashi, N. Nagashima, Y. Okamoto, S. Yamada, and T. Noguchi, "Fully compatible integration of high density embedded DRAM with 65 nm CMOS technology (CMOS5)," in *Proc. IEEE Electron Devices Meeting*, Dec. 2003, pp. 423–426.

[19] Z. Al-Ars, S. Hamdioui, and G. Gaydadjiev, "Optimizing test length for soft faults in DRAM devices," in *Proc. IEEE VLSI Test Symp.*, May 2007, pp. 59–66.

[20] S. Henzler, *Power Management of Digital Circuits in Deep Sub-Micron CMOS Technologies*. New York, NY, USA: Springer-Verlag, 2010.

[21] D.-S. Min and D. Langer, "Multiple twisted data line techniques for coupling noise reduction in embedded DRAMs," in *Proc. IEEE Custom Integr. Circuits Conf.*, May 1999, pp. 231–234.

Lifetime Enhancement Techniques for PCM-Based Image Buffer in Multimedia Applications

Yuntan Fang, Huawei Li, and Xiaowei Li

***Abstract*— Emerging nonvolatile memories such as phase change memory (PCM) have the potential to replace internal memories in embedded devices. In this brief, we propose to use PCM as image buffer in application-specific multimedia systems. To improve the lifetime of PCM-based image buffer, we first eliminate redundant writes using data comparison. After redundant write elimination, PCM cells with respect to lower order bits of pixels are written more frequently than those corresponding to higher order. Based on this observation, we show that the lifetime can be further improved either by wear leveling using periodical data reversion to make write traffic even across PCM cells or by application-level error tolerance evaluation without leveling. Experimental results demonstrate that with the proposed techniques, the lifetime of PCM-based image buffer can be improved significantly.**

***Index Terms*— Error tolerance, phase change memory (PCM), wear leveling, write traffic reduction.**

I. INTRODUCTION

Emerging nonvolatile memory (NVM) technologies have been explored as potential alternatives to SRAM, DRAM, and flash memory. They incorporate the speed of SRAM, the density of DRAM, and the nonvolatility of flash memory [1]. Phase change memory (PCM), one of the most promising NVM technologies, has the following attractive characteristics that make it appropriate as working memories: 1) high density; 2) fast startup time; 3) low idle power; and 4) a superior scalability with CMOS fabrication process. It has been explored to replace SRAM for cache [2] and DRAM for main memory [3], [4].

PCM includes two kinds of write operations, i.e., RESET (logic 0) and SET (logic 1). Although PCM has high density and low leakage power, it exhibits several write-related challenges such as limited write endurance, high write energy, and long write latency. With respect to lifetime, the typical maximum write number of one PCM cell as reported in [4] is 10^8, which is not sufficient to replace SRAM and DRAM with infinite lifetime. Once a PCM cell wears out, a fault occurs and makes PCM malfunction. Therefore, many solutions have been proposed to tackle the lifetime problem of PCM.

Prior research efforts mainly focus on exploiting emerging NVMs as main memory or cache in general computing platforms. Nevertheless, as a universal memory, NVMs also have the potential to replace internal memory components in application-specific integrated circuits (ASICs)/system-on-chips (SoCs) for embedded devices because many significant benefits of NVMs such as high density and low leakage power can be used. The characteristics of NVMs make them appropriate as on-chip/off-chip memories for data-intensive image/video applications [5], including large-capacity image buffer

Manuscript received December 13, 2012; revised March 29, 2013; accepted May 15, 2013. Date of publication August 1, 2013; date of current version May 20, 2014. This work was supported in part by the National Basic Research Program of China (973) under Grant 2011CB302501, and the National Natural Science Foundation of China under Grant 61176040 and Grant 60921002.

The authors are with the State Key Laboratory of Computer Architecture, Institute of Computing Technology, Chinese Academy of Sciences, and University of Chinese Academy of Sciences, Beijing 100190, China (e-mail: fangyuntan@ict.ac.cn; lihuawei@ict.ac.cn; lxw@ict.ac.cn).

Digital Object Identifier 10.1109/TVLSI.2013.2266668

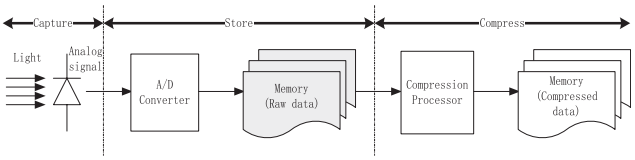

Fig. 1. Pixel architecture with internal RAM in gray [9].

in cameras [6]. Emerging high-definition images and videos require high capacity memories, e.g., the Roundshot D3 camera with a 48-bit color depth and an 80-mm lens from the Seitz company is able to shoot images of 470 million pixels [7], which means the raw image is over 2 GB. Because of its high density, PCM occupies less area and reduces memory cost. In addition to single level cells (SLC), each PCM cell can even store multiple bits, such as multiple level cells (MLC-2 and MLC-4) [8], thus achieving higher density and lower cost per bit.

In this brief, we propose to leverage NVMs as on-chip image buffer in video capture devices. Although our proposed methods do not depend on specific NVMs, for the convenience of elaboration, we take SLC PCM as an example in the rest of this brief. Fig. 1 shows a general pixel architecture containing an on-chip data memory [9], which we call image buffer. Image buffer, pervasive in digital cameras, is used to store captured raw video frames temporarily for subsequent processing, e.g., video compression. Traditionally, such a memory is usually implemented by SRAM for raw data storage. To make PCM applicable in cameras that shoot plenty of video streams, write endurance, write latency, and write energy with respect to them need to be carefully handled. We mainly concern the write endurance of PCM-based image buffer and propose specific approaches to enhance the lifetime of image buffer by using the unique feature of video applications.

We first eliminate redundant writes based on data comparison write. Afterward, a key observation is that PCM cells corresponding to most significant bits (MSBs) of pixels are written less frequently than those in regard to least significant bits (LSBs) of pixels. Hence, we propose to either reverse pixel data periodically to exchange data write locations for wear leveling or conduct error tolerance evaluation to improve the lifespan.

The rest of this brief is organized as follows. Section II surveys the related work on PCM lifetime improvement and error tolerance. Section III proposes our techniques to enhance the lifetime of PCM-based image buffer. Section IV presents our experimental results and finally we conclude our paper in Section V.

II. Related Work

Prior work on endurance enhancement of PCM is mainly for main memory and cache, for which write traffic reduction, wear leveling, and error correction are all explored.

Write traffic reduction techniques include data comparison write (DCW) [2], [3], [10], data inverting [2], [11], partial writes [4], [12], inter-block differential data encoding (IBDDE) and inter-frame multiple experts (IFME) [5], approximate write [13], and so on. DCW reduces write activity by only writing the cells whose stored value is different from the value to be written. Data inverting further reduces write traffic by calculating the hamming distance between old and new data to determine to write the original value or its inverse. Partial writes was proposed to decrease write frequency through only writing dirty lines. For video applications, IBDDE and IFME and approximate write were proposed to reduce write operations of PCM. IBDDE and IFME incur much higher logic overhead. Besides, both data inverting and IBDDE incur 12.5% storage overhead [5]. Soft PCM [13] proposed an approximate write scheme to sacrifice the data integrity for suppressing write operations. In this brief, given that DCW is simple and cost effective, we exploit it to eliminate redundant writes.

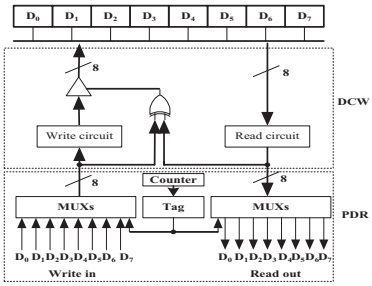

Fig. 2. Middle: DCW circuit. Bottom: PDR circuit.

To make write traffic uniform across memory cells, wear leveling techniques for PCM such as row shifting and segment swapping [3] and fine grained wear leveling [12] have been used. They were developed to mitigate the write nonuniformity and balance writes across the entire region [3] and a page [12]. Different from the previous techniques that work at the row or segment level, we propose a novel intra-pixel bit-level wear leveling technique, which is orthogonal to the previous methods for PCM.

Write traffic reduction and wear leveling strive to avoid faults, whereas error correction is used to correct errors for normal operation after hard faults occurred. Typical error-correction techniques involve ECP [14] and SAFER [15]. Error correction is not used in this brief. Instead, we use application-level error tolerance of video applications to evaluate the lifetime of PCM. As an application-specific method with no extra hardware overhead, error tolerance is much more cost effective than error-correction methods.

A lot of applications such as audios, images, and videos bear the ability of error tolerance [16]. Slight errors in them may not be perceived by human end users. Therefore, during their processing or storage, the reliability of the underlying hardware may be relaxed. Error tolerance has been used for yield improvement through threshold testing or error rate testing [17], [18] and energy efficiency [19]. In this brief, we just use error tolerance to evaluate the lifetime improvement of acceptable defective PCM in our experiments.

III. Endurance-Aware Design and Evaluation of PCM-Based Image Buffer

In this section, we make full use of the application-specific feature of image buffer to improve its lifetime. Successive frames in practical video sequences usually exhibit a high similarity. During video capture, if they are written into the same image buffer, subsequent writes into the same PCM cells have a high possibility of keeping numerical values similar to the previous written values.

A. Data Comparison Write

As shown in the middle of Fig. 2, first, we use DCW to eliminate redundant bit writes. Only changed pixel bits are written, whereas writes of unaltered pixel bits are cancelled. Compared with the baseline PCM-based image buffer where no data comparison is conducted before write, only several extra XOR gates and tristate gates are needed to achieve write reduction. Given that the read latency/power is much less than the write latency/power in PCM and the added logic gates take up a highly small portion, performance and area overhead induced by the DCW circuit can be negligible, as reported in [3] and [10]. In addition, reducing plenty of write operations, DCW helps reduce energy consumption as well.

TABLE I
LIFETIME IMPROVEMENT USING DCW

Sequence	#Frames	w_{dcw}	Improvement (%)	Sequence	#Frames	w_{dcw}	Improvement (%)
akiyo	300	176	70.5	hall	300	190	57.9
bridge-close	2001	1497	33.7	highway	2000	1088	83.8
bridge-far	2101	1267	65.8	miss-america	150	100	50.0
carphone	382	231	65.4	mobile	300	189	58.7
claire	494	289	70.9	mother-daughter	300	186	61.3
coastguard	300	187	60.4	news	300	178	68.5
container	300	181	65.7	salesman	449	449	0.0
foreman	300	184	63.0	silent	300	186	61.3
grandma	870	572	52.1	suzie	150	101	48.5
Accumulation: Total frames = 11297, w_{dcw} = 5810*, Lifetime Improvement = 94.4%							

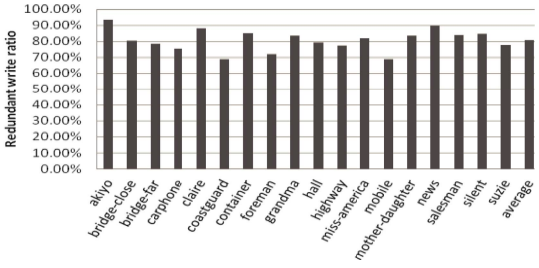

Fig. 3. Percentage of redundant writes in different video sequences.

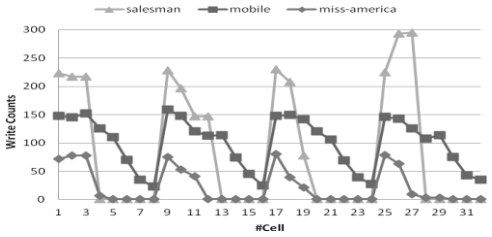

Fig. 4. Distribution of write intensities in PCM cells after DCW.

All the 18 quarter common interchange format, 176 × 144 video sequences (given in Table I) from a widely used benchmark set in the multimedia domain [20] are used to simulate the write characteristics and track the write numbers of each PCM cell accurately. Both slow- and fast-moving video sequences are included in the 18 video sequences. Through profiling them, as shown in Fig. 3, for each video sequence, abundant redundant writes exist in PCM-based image buffer. On average, ∼80% of the overall writes are redundant. Even for fast-moving sequences, such as carphone and coastguard, there are still ∼70% redundant writes.

The lifetime of PCM depends on the hottest cell. When the collective 11 297 video frames (including all the 18 video sequences) in Table I flow sequentially through the PCM-based image buffer in t s, each SLC cell in the baseline PCM will be written 11 297 times. Assume that the write endurance for each cell is equally treated as $w_{\max}$ ignoring process variations, the lifetime of the baseline PCM-based image buffer is depicted as

$$L_{\text{base}} = \frac{t * w_{\max}}{11\,297}. \tag{1}$$

We evaluate the write characteristics across different cells statistically from the 18 video sequences because different object and camera motion modes in each sequence may exercise different hot cells using DCW. After elimination of redundant writes, the write number of the hottest cell is reduced from 11 297 to some value denoted as w_{dcw} in the same period of t s. Thus, the lifetime is improved by a ratio of $(11\,297 - w_{\text{dcw}})/w_{\text{dcw}}$. Related lifetime improvement results will be analyzed in Section IV.

DCW induces an interesting result that for a group of eight cells storing pixels, cells storing MSBs of pixels usually exhibit much fewer write counts than cells storing LSBs, which is shown in Fig. 4. The horizontal axis of Fig. 4 shows 32 cells (four groups of eight cells from LSBs to MSBs). We can observe that overall this trend is stable under three different video sequences. Other video sequences show similar results. This trend holds true for both slow- and fast-moving video sequences. It is noteworthy that after applying DCW and accumulating the write counts of each cell under the 18 sequences, write intensities from LSB to MSB cells follow a perfect descending order. Such a phenomenon can be used to further improve the lifetime of PCM-based image buffer through either periodical data reversion (PDR) or error tolerance evaluation.

B. Periodical Data Reversion

As a fine-grained intrapixel wear leveling technique, PDR is proposed to mitigate the unbalanced write traffic in a row of memory cells after DCW is applied. Data reversion strives to migrate part of write activities at LSBs to MSBs periodically. The proposed data reversion circuit is shown at the bottom of Fig. 2. The hardware overhead of PDR includes a tag register, a counter, and two MUX networks at pixel write and read ports. The tag register indicates whether the written data are reversed or not. Its initial value is 0. When its value is 0, the written data remain intact. When its value is set to 1, data reversion is carried out within the MUX network, i.e., D_0 and D_7 are swapped, D_1 and D_6 are swapped, and so forth. It is noteworthy that if pixels in one image are reversely written, the pixels need to be reversed again through the MUX network at readout. The counter is used to count the reversion period. Once the counter reaches the predefined period, it is reset as 0 and the value of the tag register is inverted. The period of data reversion, expressed as the number of frames, is a key parameter. We will investigate its effect on lifetime improvement of PCM-based image buffer in Section IV. After PDR, the write number of the hottest cell, denoted as w_{pdr}, is further reduced. Hence, it can be expected to have a further improvement of the lifetime in comparison with DCW only.

C. Error Tolerance Evaluation

An alternative way to PDR is to use error tolerance evaluation (ETE) to extend the lifetime of PCM-based image buffer. Absorbing the capability of application-level error tolerance into lifetime evaluation provides a new dimension for lifespan promotion. However, it should be noted that ETE is not simultaneously used with the aforementioned PDR. That is because in PDR both MSBs and LSBs are written into each cell. If one cell fails, both MSBs and LSBs will be contaminated, which cannot be tolerated by applications.

TABLE II
FAULT INJECTION PROFILE AND LIFETIME IMPROVEMENT USING DCW AND ETE

#Faults	BER(%)	$\#D_0$	$\#D_1$	$\#D_2$	$\#D_3$	$\#D_4$	$\#D_5$	$\#D_6$	$\#D_7$	w_{relax}	Improvement (%)
10034	4.9	9793	241	0	0	0	0	0	0	5413	108.7
20004	9.9	16909	3095	0	0	0	0	0	0	5175	118.3
30025	14.8	20243	9768	1	0	0	0	0	0	4925	129.4
40004	19.7	21108	18657	239	0	0	0	0	0	4532	149.3
50022	24.7	21817	21129	7070	6	0	0	0	0	3982	183.7
60003	29.6	23629	21358	14979	37	0	0	0	0	3595	214.2
70007	34.5	25199	23455	20279	1067	7	0	0	0	3071	267.9

TABLE III
APPLICATION-LEVEL VIDEO QUALITY EVALUATION

Sequence	#Faults	SSIM1	SSIM2	SSIM3	CRR	Sequence	#Faults	SSIM1	SSIM2	SSIM3	CRR
akiyo	10034	0.9706	**0.9989**	0.9711	0	foreman	10034	0.9561	**0.999**	0.9563	0
	20004		**0.9971**	0.9709	0		20004		**0.9972**	0.9563	0.01
	30025		**0.9932**	0.9707	0		30025		**0.9946**	0.9567	0.01
	40004		**0.9869**	0.9709	0.03		40004		**0.9908**	0.9568	0.03
	50022		**0.976**	0.9709	0.06		50022		**0.9815**	0.9572	0.06
	60003		0.9605	**0.9695**	0.09		60003		**0.9705**	0.958	0.09
	70007		0.9433	**0.9672**	0.14		70007		0.9579	**0.9585**	0.14
coastguard	10034	0.9262	**0.9991**	0.9261	0	hall	10034	0.9718	**0.9987**	0.972	0
	20004		**0.9982**	0.9262	0.01		20004		**0.9971**	0.9722	0.02
	30025		**0.9954**	0.9265	0.02		30025		**0.9928**	0.9722	0.02
	40004		**0.9927**	0.9269	0.03		40004		**0.9875**	0.972	0.02
	50022		**0.9836**	0.9284	0.08		50022		**0.9726**	0.9715	0.06
	60003		**0.974**	0.9296	0.13		60003		0.9587	**0.9701**	0.07
	70007		**0.9659**	0.9304	0.18		70007		0.9445	**0.9691**	0.11

When cell faults only contaminate LSBs, error tolerance can be readily used. From Section III-A, we have known that PCM cells storing LSBs of pixels experience much higher write intensities after redundant write elimination via DCW. According to such a trend, hard faults will first occur in cells with respect to LSBs. If the hard faults have slight effects on video quality, we deem that the defective PCM-based image buffer can still work and continue to extend its effective lifetime. Combing DCW and ETE, image buffer works normally in its early life and experiences incremental but acceptable quality degradation in its lifetime because of the gradually increasing number of wearout PCM cells.

After applying ETE, the choice of the hottest cell in the PCM can be relaxed. For example, to capture a sequence of 1000 frames, after we eliminate redundant writes, the hottest cell C_1 (storing an insignificant bit) is to be written 800 times and the next hottest cell C_2 400 times. If subsequent write activities of C_1 and C_2 follow a similar tendency during the traditional lifetime limit 10^8, when C_1 has been written 10^8 times, C_2 will have been written 5×10^7 times. Thereafter, C_1 is faulty but C_2 still has another 5×10^7 write times before it is faulty. If the faulty cell $C1$ has no obvious effect on the final outcome, PCM can, however, still function as before. Thus, the effective lifetime of PCM-based image buffer is prolonged. We use w_{relax} to represent the write number of the relaxed hottest cell (e.g., C_2 in the above example) after excluding much hotter but inessential cells. It is less than the write number of the truly hottest cells (e.g., C_1 in the above example) and depends on the number of uncritical cells related with LSBs.

In this brief, we conduct fault injection experiments to simulate the condition that after a maximum write number is consumed for some hottest cells, hard faults have occurred in the PCM-based image buffer. With redundant writes removed, faults first occur in the cells with the largest write numbers and then spread toward cells with less write numbers progressively. Once a cell is faulty, cells that are hotter than it must have been faulty. Hence, we inject faults into memory cells based on the descending order of write numbers of all PCM cells. For the PCM cells with the same write number, we always inject faults into them simultaneously. We inject either a stuck-at-0 called RESET or stuck-at-1 called SET fault into each faulty PCM cell randomly.

After fault injection, we evaluate the fault effects on video quality, compression ratio, and video encoding/decoding time. Structural similarity (SSIM) proposed in [21] is used to evaluate video quality. SSIM of one video sequence is the average of SSIMs of all video frames. For two identical video sequences, SSIM is 1. SSIM is < 1 in case of quality degradation.

IV. EXPERIMENTAL RESULTS

A. Lifetime Enhancment

In this section, we present lifetime improvement results with pure DCW, DCW and PDR, and DCW and ETE, respectively. Because we mainly focus on write endurance of PCM-based image buffer, we build an efficient system-level simulator that integrates a list of functions for video preprocessing, DCW, PDR, fault injection, and ETE. In addition, we use JM 9.4 [22] for video encoding/decoding. JM 9.4 is a reference model for H.264/AVC video encoders/decoders, where the default baseline configuration is used in the experiments. As shown in Table I, the video test sequences that are widely used in the multimedia domain [20] are employed to simulate the video capture process. Grayscale video frames are considered in the experiments. Irrespective of image types, our techniques are, however, applicable to both grayscale images and color ones given that they bear intrinsic interframe redundancy.

Table I lists lifetime improvement results using DCW in the fourth and eighth columns for each video. The w_{dcw} is the write number of the hottest cell. Both the hottest cell and lifetime improvement vary under different video sequences because they have diverse object and camera motion modes and asymmetric inter-frame redundancy. The write numbers of the hottest cell with and without DCWs are the same for saleman, which leads to no improvement. The last row in Table I lists the accumulated lifetime improvement. It is noteworthy that the complementary effects among different video sequences (different

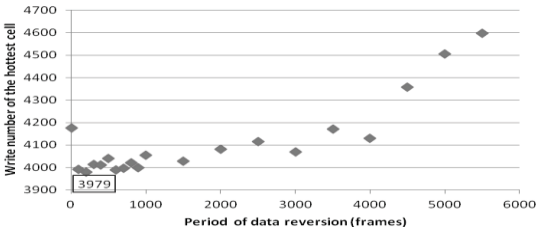

Fig. 5. Effect of reversion period on the write number of the hottest cell.

TABLE IV
PCM CELL PARAMETERS UNDER 90-nm TECHNOLOGY [4]

Cell Size	$0.097\mu m^2$
Array Size	202752x0.097 = $19666.944\mu m^2$
Access Device	BJT
Read Operation	Latency: 48ns, Power: 40uW, Energy: 2.0pJ
Set Operation	Latency: 150ns, Power: 90uW, Energy: 13.5pJ
Reset Operation	Latency: 40ns, Power: 480uW, Energy: 19.2pJ

TABLE V
AREA, PERFORMANCE, AND POWER OVERHEAD OF DCW AND PDR

Circuit	Overhead (90nm)		
	Area (μm^2)	Delay (ns)	Power (uW)
DCW	95.96	0.10	11.00
PDR	344.33	0.54	21.05
ETE	No hardware overhead		

video sequences exhibit different most written PCM cells) make the write number (5810, asterisked in Table I) of the hottest cell in the accumulated 18 sequences less than the sum of w_{dcw} for the total 18 sequences. Therefore, the accumulated lifetime improvement, i.e., 94.4%, is higher than any improvement under one single video sequence or the average improvement. We believe that this result reflects the actual write behavior in a statistical sense, whereas the average result underestimates the lifetime. Afterward, we evaluate lifetime enhancement based on the collective write characteristic of the 18 sequences.

The reversion period of PDR is a key parameter for wear leveling. Fig. 5 shows the effect of the data reversion period on the write number of the hottest cell. With a reversion period of 200 frames, w_{pdr} will equal 3979, and the lifetime improvement can be up to 183.9% after combining DCW and PDR. Compared with pure DCW, PDR provides an additional 89.5% lifetime improvement that indicates a great benefit of combining write traffic reduction and wear leveling techniques.

Table II lists a fault injection profile and the corresponding lifetime improvement via ETE if the faults that occurred in PCMs do not cause intolerable errors in the application, which is listed in the last column. The first column is the total number of injected faults and the second represents the bit error rate (BER, i.e., the percentage of faulty cells). Columns 3–10 indicate the numbers of faults distributed in cells storing LSBs ($\#D_0$) to MSBs ($\#D_7$). The w_{relax} in the 11th column is the largest cell write number after excluding all the faulty cells, which refers to the write number of the hottest fault-free cell. Application-level video quality evaluation verifies whether injected faults affect video quality significantly or not. Video quality before compression and that after compression are both considered. The evaluation results are given in Table III. Taking the original error-free video as reference, SSIM1, SSIM2, and SSIM3 represent the quality of the compressed version of the error-free video, the contaminated video, and the compressed counterpart of the contaminated video, respectively. Concerning the encoding/decoding time of contaminated video against that of the error-free video, faulty PCM cells almost have no effects. The compression ratio reduction (CRR) is also considered. When even up to 50 020 or 60 003 faults are injected into the PCM-based image buffer, both the quality of contaminated video and that of its reconstructed counterpart are still good with respect to the original uncompressed video, only incurring slight SSIM differences. In addition, another marginal overhead is CRR of the contaminated video sequences, which is $< 10\%$ on average. This means that contaminated videos become less compressible compared with the original videos. On the one hand, such an overhead is acceptable because external storage is usually abundant and much cheaper. On the other hand, because of the small loss of compression ratio, sometimes the quality of the compressed version of the slightly contaminated video is even better than that of the compressed one of the error-free video. What is more interesting, when the captured video is slightly affected, the quality of the contaminated video without compression is better than its compressed version; however, when the captured video is more seriously affected by the faulty cells, its quality may be a little lower than that of its compressed counterpart, e.g., for hall video sequence, when 60 003 faults are injected, SSIM3 (equal to 0.9701) is higher than SSIM2 (equal to 0.9587). That is because in video compression, although quantization after 2-D discrete cosine transformation usually incurs quality loss, it may also help to filter some high-frequency components induced by stuck-at faults in the PCM, thus even improving a little video quality after compression. Tables II and III show that lifetime improvement after DCW and ETE techniques are combined together, can reach up to 183.7% or 214.2%, when 50 022 or 60 003 faults occur in the PCM (with BER equal to 24.7% or 29.6%).

B. Overhead in Area, Delay, and Power

We include all the hardware overhead induced by DCW and PDR for evaluation. The bit-width of the counter in PDR is 8 because the reversion period is 200 frames. The inserted logic was synthesized using Synopsys Design Compiler with SMIC 90-nm technology to evaluate area, delay, and power overhead. We obtain the reference parameters with respect to PCM under 90-nm technology from [4] for comparison. They are listed in Table IV. Table V shows the overhead of DCW and PDR in terms of area, delay, and power. Compared with the area of 202 752 ($176 \times 144 \times 8$) PCM cells, the total area of DCW and PDR is $\sim$2% of the baseline PCM-based image buffer. Furthermore, practical video sequences with a resolution much higher than 176×144 enable the size of image buffer to be larger, which makes the area overhead of DCW and PDR reasonably negligible. The delay and power incurred by DCW and PDR for writing/reading a pixel are also small. Although DCW needs an extra read operation, it eliminates 80% writes and thus saves much more power than the additional read power it consumes.

V. CONCLUSION

In this brief, PDR and ETE are proposed to enhance the lifetime of PCM-based image buffer after data comparison write. Data comparison first eliminates inherent redundant writes by taking advantage of temporal redundancy. PDR exchanges data write locations for wear leveling, whereas ETE extends the effective lifetime with a slight loss of video quality and compression ratio. Compared with the baseline PCM-based image buffer, both DCW and PDR and DCW and ETE can achieve lifetime improvement by a factor of 1.84, with negligible area, performance, and power overhead.

REFERENCES

[1] Y. Xie, "Modeling, architecture, and applications for emerging memory technologies," *IEEE Design Test Comput.*, vol. 28, no. 1, pp. 44–50, Jan.–Feb. 2011.

[2] Y. Joo, D. Niu, X. Dong, G. Sun, N. Chang, and Y. Xie, "Energy- and endurance-aware design of phase change memory caches," in *Proc. Design, Autom. Test Eur. Conf. Exhibit.*, Mar. 2010, pp. 136–141.

[3] P. Zhou, B. Zhao, J. Yang, and Y. Zhang, "A durable and energy efficient main memory using phase change memory technology," in *Proc. Int. Symp. Comput. Archit.*, 2009, pp. 14–23.

[4] B. C. Lee, E. Ipek, O. Mutlu, and D. Burger, "Architecting phase change memory as a scalable DRAM alternative," in *Proc. Int. Symp. Comput. Archit.*, 2009, pp. 2–13.

[5] S. Kwon, S. Yoo, S. Lee, and J. Park, "Optimizing video application design for phase-change RAM-based main memory," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 20, no. 11, pp. 2011–2019, Nov. 2012.

[6] (2013). *Buffer* [Online]. Available: http://www.dpreview.com//learn/?/Glossary/Camera_System/buffer_01.htm

[7] (2013). *470 Megapixel Seitz Roundshot D3* [Online]. Available: http://www.photographyblog.com/news/470_megapixel_seitz_roundshot_d3

[8] T. Nirschl, J. B. Philipp, T. D. Happ, G.W. Burr, "Write strategies for 2 and 4-bit multi-level phase-change memory," in *Proc. IEDM*, 2007, pp. 461–464.

[9] M. Zhang and A. Bermak, "Quadrant-based online spatial and temporal compressive acquisition for CMOS image sensor," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 19, no. 9, pp. 1525–1534, Sep. 2011.

[10] B. D. Yang, J. E. Lee, J. S. Kim, J. Cho, S. Y. Lee, and B. G. Yu, "A low power phase-change random access memory using a data-comparison write scheme," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2007, pp. 3014–3017.

[11] S. Cho and H. Lee, "Flip-N-Write: A simple deterministic technique to improve PRAM write performance, energy and endurance," in *Proc. 42nd Annu. IEEE ACM Int. Symp. Microarchitecture*, Dec. 2009, pp. 347–357.

[12] M. K. Qureshi, V. Srinivasan, and J. A. Rivers, "Scalable high performance main memory system using phase-change memory technology," *ACM SIGARCH Comput. Archit. News*, vol. 37, no. 3, pp. 24–33, 2009.

[13] Y. Fang, H. Li, and X. Li, "SoftPCM: Enhancing energy efficiency and lifetime of phase change memory in video applications via approximate write," in *Proc. Asian Test Symp.*, 2012, pp. 131–136.

[14] S. Schechter, G. H. Loh, K. Strauss, and D. Burger, "Use ECP, not ECC, for hard failures in resistive memories," in *Proc. 37th Annu. Int. Symp. Comput. Architecture*, 2010, pp. 141–152.

[15] N. H. Seong, D. H. Woo, V. Srinivasan, J. A. Rivers, and H. H. S. Lee, "SAFER: Stuck-at-fault error recovery for memories," in *Proc. IEEE/ACM Int. Symp. Microarchitecture*, Jan. 2010, pp. 115–224.

[16] Y. Fang, H. Li, and X. Li, "A fault criticality evaluation framework of digital systems for error tolerant video applications," in *Proc. Asian Test Symp.*, 2011, pp. 329–334.

[17] Z. Jiang and S. K. Gupta, "Threshold testing: Improving yield for nanoscale VLSI," *IEEE Trans. Comput. Aided Design Integr. Circuits Syst.*, vol. 28, no. 12, pp. 1883–1895, Dec. 2009.

[18] K.-J. Lee, T.-Y. Hsieh, and M. A. Breuer, "A novel test methodology based on error-rate to support error tolerance," in *Proc. IEEE Int. Test Conf.*, Nov. 2005, pp. 1136–1144.

[19] M. Cho, J. Schlessman, W. Wolf, and S. Mukhopadhyay, "Reconfigurable SRAM architecture with spatial voltage scaling for low power mobile multimedia applications," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 19, no. 1, pp. 161–165, Jan. 2011.

[20] (2006). *YUV Video Sequences* [Online]. Available: http://trace.eas.asu.edu/yuv/index.html

[21] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[22] (2008). *JM 9.4 (JM Reference Software for H.264/AVC)* [Online]. Available: http://iphome.hhi.de/suehring/tml/index.htm

Nonbinary LDPC Decoder Based on Simplified Enhanced Generalized Bit-Flipping Algorithm

Francisco García-Herrero, María José Canet, and Javier Valls

***Abstract*— A simplified version of the enhanced serial generalized bit-flipping algorithm is proposed in this brief. This new algorithm reduces the quantity of information that is stored with a negligible performance loss of 0.05 dB compared with previous proposals. In addition, the algorithm allows us not only to save memory, but also to reduce the number of arithmetic resources needed. In addition, a new initialization of the algorithm avoids using techniques to control data growth without any performance degradation, reduces routing, increasing the maximum frequency achievable, and saves logic. The decoder derived from the simplified algorithm requires almost half the area of previous architectures, with a throughput of 716 Mbps on a 90-nm CMOS process for the (837, 723) nonbinary code over GF(32) at ten iterations.**

***Index Terms*— Error correction codes, hardware architecture, iterative decoding, nonbinary low-density parity-check codes, symbol flipping decoding.**

I. INTRODUCTION

The design of nonbinary low-density parity-check (NB-LDPC) decoder architectures with moderated area and high throughput remains a challenge for very large scale integration designers. Decoding algorithms derived from q-ary sum-product algorithm [1]–[4] as extended min-sum (EMS) [5], simplified min-sum (SMS) [6], and min-max (MM) [7] involve high complexity in their check node update (CNU) rules. The CNU processing of EMS (in all its variants [8]–[10]), SMS, and MM requires the search of the symbols that satisfy the parity check equations, making use of the soft information from the channel. This search can be exhaustive or not, but its implementation always limits the throughput of the decoder architectures. On the one hand, exhaustive search techniques, such as forward–backward (FB), imply all to all comparisons [11]. This process is recursive and requires forward, backward, and merge steps that perform $p \times p$ comparisons in each check node, where p is the size of the Galois field (GF). On the other hand, nonexhaustive search is usually based on trellis [12], which can be simplified by reducing the number of paths. In this way, the amount of storage resources is lower than in FB, but the data dependency penalizes the latency of the decoder. In the current state of the art, architectures for EMS, SMS, or MM do not reach a throughput of 100 Mbps [11]–[17].

If higher throughput and lower area are required, there are two main solutions, to reduce the complexity of the CNU processing or decrease the number of iterations of EMS, SMS, or MM. To reduce the complexity of the CNU processing, hard decision solutions can be applied to select d_c candidates that satisfy the check node equations. This approach is followed by the majority-logic

Manuscript received December 26, 2012; revised May 21, 2013; accepted July 25, 2013. Date of publication August 21, 2013; date of current version May 20, 2014. This work was supported in part by the Spanish Ministerio de Ciencia e Innovación under Grant TEC2011-27916 and in part by the Universidad Politécnica de Valencia under Grant PAID-06-2012-SP20120625. The work of F. García-Herrero was supported by the Spanish Ministerio de Educación under Grant AP2010-5178.

The authors are with the Instituto de Telecomunicaciones y Aplicaciones Multimedia, Universidad Politecnica de Valencia, Gandia 46730, Spain (e-mail: fragarh2@upv.es; macasu@upv.es; jvalls@upv.es).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVLSI.2013.2276067